L11: Entry 6 of 8 File: PGPB Feb 28, 2002

DOCUMENT-IDENTIFIER: US 20020024517 A1
TITLE: Apparatus and method for three-dimensional image production and presenting real objects in virtual three-dimensional space

Abstract Paragraph:
Using a stereo viewing method, three-dimensional model data are produced that completely express an object in a three-dimensional shape, or moving images of the object as seen from any viewpoint are produced.

Summary of Invention Paragraph:
[0002] This invention relates to an apparatus and method for producing three-dimensional model data for an object, or producing images that view the object from any viewpoint, based on object distance data obtained by a stereo ranging method. And this invention relates to a system and method for presenting three-dimensional model data for real objects in virtual three-dimensional space.

Summary of Invention Paragraph:
[0005] With the prior art described above, curved-surface shapes of surfaces of an object seen from individual observation locations are computed, but three-dimensional model data that completely represent the three-dimensional shape of the object are not produced.

Summary of Invention Paragraph:
[0007] Why then, systems that involve computer-based virtual three-dimensional space are being proposed for various applications such as apparel trial fitting and direct-involvement games and the like. In Japanese Patent Application Laid-Open No. H10-124574/1998, for example, a system is disclosed that is made so that three-dimensional model data are produced for a user's body from photographs of and/or dimensional data on the user's body, that user's body three-dimensional model is imported into the virtual three-dimensional space of a computer, and, by clothing that model with three-dimensional apparel models and applying lipstick color data and the like, apparel and lipstick try-on simulations can be done. Analogous or related trial fitting systems are disclosed in Japanese Patent Application Laid-Open No. H10-340282/1998, H11-203347/1999, and H11-265243/1999, etc.

Summary of Invention Paragraph:
[0009] In the conventional game systems described above, three-dimensional models of appearing characters existing in virtual three-dimensional space have a certain form that is altogether unrelated to the physical characteristics of the game player. In that regard, the reality level is still unsatisfactory in the sense of the player himself or herself becoming a character appearing in the game. On the other hand, in the conventional trial fitting systems described in the foregoing, three-dimensional model data of the user's body are imported into virtual three-dimensional space, wherefore the reality of the user himself or herself doing the trying on is very high.

Summary of Invention Paragraph:
[0010] Nevertheless, the prior art described in the foregoing does not provide any specific method or means for producing three-dimensional model data of the user's body. If, in order to produce three-dimensional model data of the user's body, the user himself or herself must have very expensive equipment, or enormous time and effort or costs are involved, then it will be very difficult to render practical a system that uses virtual space, such as described in the foregoing.

Summary of Invention Paragraph:
[0011] Accordingly, one object of the present invention is to make it possible to produce

three-dimensional model data that completely represent the three-dimensional shape of an object, using a stereo ranging method.

Summary of Invention Paragraph:
[0013] Another object of the present invention is to generate three-dimensional model data of such real physical objects as a person's body or article, without placing an overly large burden on the user, and to make provision for that three-dimensional model to be imported into virtual three-dimensional space.

Summary of Invention Paragraph:
[0014] Another object of the present invention is to make provision so that, in order to further enhance the reality of the virtual three-dimensional space into which three-dimensional model data for a real object has been imported, those three-dimensional model data can be made to assume different poses and perform motion inside the virtual three-dimensional space.

Summary of Invention Paragraph:
[0021] According to a third perspective of the present invention, a system for making it possible to cause a real physical object to appear in virtual three-dimensional space in a computer application used by a user that follows a first perspective of the present invention comprises: photographed data reception means for receiving photographed data produced by stereo photographing a real physical object, from a stereo photographing apparatus usable by the user, capable of communicating with that stereo photographing apparatus; modeling means for producing a three-dimensional model of the physical object, based on the received photographed data, in a prescribed data format that can be imported into virtual three-dimensional space by the computer application; and three-dimensional model output means for outputting the produced three-dimensional model data by a method wherewith those data can be presented to the user or a computer application used by the user.

Summary of Invention Paragraph:
[0022] If this system is used, if a user photographs a physical object, such as his or her own body or an article, which he or she wishes to import into the virtual three-dimensional space of a computer application, with a stereo photographing apparatus, and transmits those photographed data to this system, three-dimensional model data for that physical object can be received from this system, wherefore the user can import those received three-dimensional model data into his or her computer application.

Summary of Invention Paragraph:
[0023] In a preferred embodiment aspect, this system exists as a modeling server on a communications network such as the internet. Thereupon, if a user photographs a desired physical object with a stereo photographing apparatus installed in a store such as a department store, game center, or convenience store or the like, for example, or with a stereo photographing apparatus possessed by the user himself or herself, and transmits those photographed data to a modeling server via a communications network, that three-dimensional model will be sent back via the communications network to the computer system in the store or to the computer system in the possession of the user. Thus the user can easily access a three-dimensional model of a desired physical object, and import that into a desired application such as a virtual trial fitting application or direct-involvement game or the like.

Summary of Invention Paragraph:
[0024] In a preferred embodiment aspect, the photographed data for a physical object photographed with a stereo photographing apparatus comprises photographed data of a plurality of poses photographed when that physical object assumed respectively different poses. For example, if the stereo photographing apparatus employs a video camera, when the user photographs himself or herself, for example, if that photographing is done while various poses are assumed or motions are performed, photographed data for many different poses will be obtained. The modeling means receive the photographed data for such different poses and, based thereon, produce three-dimensional model data of a configuration wherewith different poses can be assumed and motions performed. The user, thereby, can import the produced three-dimensional model data into the virtual three-dimensional space of a computer application, and then cause that three-dimensional model to assume various different poses or perform motions.

Summary of Invention Paragraph:
[0027] A system that follows a fifth perspective of the present invention further combines, in

addition to the stereo photographing apparatus and modeling apparatus described in the foregoing, a computer apparatus capable of executing a computer application that imports produced three-dimensional models into virtual three-dimensional space.

Detail Description Paragraph:
[0205] With the arithmetic logic units 18, 200, 300, and 400 diagrammed in FIG. 2, 4, 5, and 6, on the other hand, a complete three-dimensional model of the object 10 is produced by the modeling and display unit 78 (a three-dimensional model which in fact moves so as to follow the motion of the object 10 in real time), wherefore it is possible to make the configuration such that that three-dimensional model is extracted and imported to another graphic processing apparatus (such as a game program for performing computer three-dimensional animation). When that is done, applications are possible wherewith the three-dimensional model of the object 10 is displayed moving in another graphic processing apparatus (such, for example, as applications that import the three-dimensional model of a real game player that is the object 10 into the game program noted above, such that that three-dimensional model takes part in the virtual world displayed by that game program while moving in the same way as the game player).

Detail Description Paragraph:
[0209] The store system 1005 takes the photographed data of the user's body received from the stereo photographing system 1006 and sends them to the modeling server 1001 via the communications network 1008. The modeling server 1001 produces three-dimensional modeling data for the user's body, using the photographed data of the user's body received from the store system 1005, by performing processing that will be described in detail subsequently with reference to FIG. 16 to 20. The modeling server 1001 stores the produced three-dimensional model data of the user's body in a user database 1001A, and then transmits those three-dimensional model data of the user's body via the communications network 1008 to the store system 1005. The store system 1005 sends those three-dimensional model data of the user's body via the communications network 1008 (or via a transportable recording medium such as a recording disk) to the user system 1004. Or, alternatively, provision may be made so that the modeling server 1001, when so requested by the user system 1004, transmits the three-dimensional model data of the user's body stored in the user database 1001A directly to the user system 1004 via the communications network 1008.

Detail Description Paragraph:
[0211] Now, the apparel supplier system 1002 produces three-dimensional model data of various apparel items (clothing, shoes, hats, accessories, bags, etc.) supplied by that apparel supplier, accumulates those data in the apparel database 1002A, and sends those apparel three-dimensional model data to the virtual trial fitting server 1003 via the communications network 1008 or via a disk recording medium or the like. Alternatively, the apparel supplier system 1002 may photograph apparel (or a person wearing that apparel) with a stereo photographing system that is the same as or similar to the stereo photographing system 1006 of the store, send those photographed data to the modeling server 1001, and have the modeling server 1001 produce three-dimensional model data for that apparel, then have the three-dimensional model data for that apparel received from the modeling server 1001 and sent to the virtual trial fitting server 1003 (or, alternatively, have those data sent directly from the modeling server 1001 to the virtual trial fitting server 1003 via the communications network 1008).

Detail Description Paragraph:
[0212] The virtual trial fitting server 1003 might be the website of a department store or clothing store, for example. Thereupon, three-dimensional model data of various apparel items received from the apparel supplier system 1002, etc., are accumulated in the apparel database 1003A supplier by supplier, or there is a virtual trial fitting program 1003B that can be run on the user system 1004. Then, when requested by the user system 1004, the virtual trial fitting server 1003 sends the three-dimensional model data for those various apparel items and the virtual trial fitting program to the user system 1004 via the communications network 1008.

Detail Description Paragraph:
[0213] The user system 1004 installs the three-dimensional model data of the user's body received from the modeling server 1001, and the three-dimensional model data for the various apparel items and virtual trial fitting program received from the virtual trial fitting system 1003 on a hard disk drive or other auxiliary memory device 1004A, and then runs the virtual trial fitting program according to the directions of the user. The three-dimensional model data of the user's body and the three-dimensional apparel model data are made in a prescribed data

format that can be imported into the virtual three-dimensional space by the virtual trial fitting program. The virtual trial fitting program imports the three-dimensional model data of the user's body and the three-dimensional model data for various apparel into the virtual three-dimensional space, dresses the three-dimensional model of the user with preferred apparel, causes preferred poses to be assumed and preferred motion to be performed, renders images of that figure as seen from preferred viewpoints, and displays those images on a display screen. The virtual trial fitting program, moreover, by using known art to map any color or texture to any site in the three-dimensional model data of the user's body or apparel, can simulate appearances in various cases, such as when the model has been suntanned, or has put on various kinds of cosmetics, or has dyed his or her hair, or has changed the color of his or her clothes, etc. Or, using known art to subject the three-dimensional model data of the user's body to enlargement, reduction, deformation, or replacement with another model, appearances can be simulated such as when the model has become heavier, has become thinner, has grown in stature, or has altered his or her hair style, etc. The virtual trial fitting program can also accept orders for any apparel from the user and send those orders to the virtual trial fitting server 1003.

Detail Description Paragraph:
[0214] According to this virtual trial fitting system, the user, even though not having his or her own equipment for three-dimensional modeling, nevertheless can, by going to a department store, game center, or convenience store and photographing his or her own body with the stereo photographing system 1006 installed there, have three-dimensional model data of his or her own body made, import those data into his or her own computer, and, using those three-dimensional model data for himself or herself, try on various apparel items at a high reality level in the virtual three-dimensional space of the computer. In addition, as will be described subsequently, it is possible to use those three-dimensional model data of himself or herself not only for virtual trial fitting, but also by importing and using those data in the virtual three-dimensional space of direct-involvement games and other applications. Also, if user photographed data or three-dimensional model data based thereon are acquired and employed, with the consent of the user and in a way that does not infringe on the privacy of the user, it becomes possible to design and manufacture apparel ideally suited to the body of the user, at lower than conventional cost, or to develop and design new apparel that is more advanced in terms of human engineering, based on detailed data on the human body not obtainable by ordinary measurement taking.

Detail Description Paragraph:
[0215] FIG. 9 and FIG. 10 represent the processing procedures for this virtual trial fitting system in greater detail. FIG. 9 represents processing procedures for producing three-dimensional model data for a user's body performed centrally by the modeling server 1001. FIG. 10 represents processing procedures for performing virtual trial fitting on a user system, centrally by the virtual trial fitting server 1003.

Detail Description Paragraph:
[0216] First, the processing procedures for producing three-dimensional model data for a user's body are described with reference to FIG. 8 and FIG. 9.

Detail Description Paragraph:
[0220] (4) The modeling server 1001, based on the received full-body photographed data, produces three-dimensional physique model data representing the full-body shape of the user (S1003).

Detail Description Paragraph:
[0222] (6) The modeling server 1001, based on the local part photographed data received, produces three-dimensional local part model data that represents the shape of the local parts, particularly the face, of the user (S1005).

Detail Description Paragraph:
[0223] (7) The modeling server 1001, by inserting the corresponding three-dimensional local part model data into the face and other local parts of the three-dimensional physique model data for the full body produces a standard full-body model that represents both the shape of the full body of the user and the detailed shapes of the face and other local parts (S1006). The modeling server 1001 transmits that standard full-body model to the store system 1005 (S1007), and the store system 1005 receives that standard full-body model (S1014).

Detail Description Paragraph:
[0227] (1) The apparel supplier system 1002 produces three-dimensional model data for various apparel items (S1031), and transmits those data to the virtual trial fitting server 1003 (S1032). The virtual trial fitting server 1003 receives those three-dimensional model data for the various apparel items and accumulates them in the apparel database (S1041).

Detail Description Paragraph:
[0228] (2) The user system 1004 requests access to the virtual trial fitting server 1003 at any time (S1051). The virtual trial fitting server 1003, upon receiving the request for access from the user system 1004 (S1042), transmits the virtual trial fitting program and the three-dimensional model data for the various apparel items to the user system 1004 (S1043). The user system 1004 installs the virtual trial fitting program and the three-dimensional model data for the various apparel items received in its own machine so that it can execute the virtual trial fitting program (S1052). Furthermore, there is no reason why the virtual trial fitting program and the three-dimensional apparel model data must always be downloaded from the virtual trial fitting server 1003 to the user system 1004 simultaneously. The virtual trial fitting program and the three-dimensional apparel model data may be downloaded on different occasions, or, alternatively, either one or other or both of the virtual trial fitting program and the three-dimensional apparel model data may be distributed to the user, not via a communications network, but recorded on a CD-ROM or other solid recording medium and installed in the user system 1004.

Detail Description Paragraph:
[0245] Next, as indicated in step S1062, the modeling server produces, from those photographed data for the differing plurality of poses, three-dimensional model data for the full-length physique of the user for each pose. The three-dimensional physique model data for each pose produced at that time constitute three-dimensional model data that capture the full body of the user as one cubic body (hereinafter called the full-body integrated model), as indicated by the reference number 1600 in FIG. 13.

Detail Description Paragraph:
[0250] As indicated in FIG. 14, the virtual trial fitting program, in step S1071, obtains the standard full-body model 1601 for the user. Also, in step S1072, the virtual trial fitting program obtains the three-dimensional model data for the apparel selected by the user. These three-dimensional apparel model data, as indicated by the reference number 1620 in FIG. 15, are divided into a plurality of parts 1621 to 1627 in the same manner as the standard full-body model of the user, and those parts 1621 to 1627 are configured such that they are articulated by joints indicated by black dots.

Detail Description Paragraph:
[0251] Next, as indicated in step S1073, the virtual trial fitting program positions the three-dimensional model data for the apparel to (that is, places the apparel on) the standard full-body model 1601 of the user in the virtual three-dimensional space.

Detail Description Paragraph:
[0252] Next, as indicated in step S1074, the virtual trial fitting program progressively deforms the standard full-body model 1601 and the three-dimensional apparel model data 1620, bending them at the joints, so that the standard full-body model 1601 wearing the apparel assumes the poses and performs the motions designated by the user, as indicated by the reference number 1630 in FIG. 15, in the virtual three-dimensional space. Then, as indicated in step S1075, two-dimensional images of the standard full-body model 1601 and the three-dimensional apparel model data 1620 that progressively deform in that manner are rendered, as seen from a user-designated camera position and user-designated zoom magnification, are rendered and displayed on the show stage window 1501 indicated in FIG. 11.

Detail Description Paragraph:
[0255] The output signals from the multi-eyes stereo cameras 1011, 1012, and 1013 are input to the arithmetic logic unit 1018. The arithmetic logic unit 1018 produces three-dimensional model data for the object 1010, based on the signals input from the multi-eyes stereo cameras 1011, 1012, and 1013. Here, the arithmetic logic unit 1018 is represented in the drawing as a single block for convenience, but connotes the functional components that perform three-dimensional modeling, formed by the combination of the virtual trial fitting system and store system 1005

diagrammed in FIG. 8.

Detail Description Paragraph:
[0377] In FIG. 22 is diagrammed the overall configuration of one embodiment aspect of a game system that follows the present invention. This game system is for a user to import three-dimensional model data of any physical object into the virtual three-dimensional space of a computer game and play therewith.

Detail Description Paragraph:
[0382] (3) As indicated in steps S1101 and S1091, the modeling server 1701 receives and accumulates information representing the data format for the three-dimensional models used by various game programs, beforehand, from the game supplier system 1702. The modeling server 1701 then, after receiving a game ID and photographed data for an item 1709 from the user system 1704, thereupon, as indicated in step S1093, produces three-dimensional model data for that item 1709, in the data format for the game ID received, using the photographed data received. The way in which the three-dimensional model data are made is basically the same as the method described with reference to FIG. 17 to 20. The modeling server 1701 transmits the three-dimensional model data produced for the item to the user system 1704 and, as indicated in step S1083, the stereo photographing program in the user system 1704 receives those three-dimensional model data.

Detail Description Paragraph:
[0383] (4) As indicated in step S1083, the stereo photographing program in the user system 1704, using the three-dimensional model data received, renders that three-dimensional model into two-dimensional images as seen from various directions (into moving images seen from all directions while turning that three-dimensional model, for example), and displays those on the display device 1707. The user views those images to check whether there are any problems with the received three-dimensional model data. When it has been verified that there are no such problems, the stereo photographing program stores the received three-dimensional model data, and notifies the modeling server 1701 that receipt has been made.

Detail Description Paragraph:
[0384] (5) As indicated in steps S1094 and S1095, the modeling server 1701, when it has been verified that the user has received the three-dimensional model data, performs a fee-charging process for collecting a fee from the user, transmits data resulting from that fee-charging process such as an invoice to the user system 1704, and, as indicated in step S1085, the stereo photographing program in the user system 1704 receives and displays those data resulting from that fee-charging process.

Detail Description Paragraph:
[0385] (6) After the production of the three-dimensional model data for the item 1709 has been finished in this manner, then, as indicated in step S1086, the user runs the game program on the user system 1704 and, in that game program, the three-dimensional model data for the item 1709 stored earlier are used. For example, as illustrated in the display device 1707 in FIG. 22, the user can use the three-dimensional model 1708 of his or her toy automobile 1709 and play the automobile race game.

Detail Description Paragraph:
[0386] FIG. 25 represents a second embodiment aspect of a game system that follows the present invention. This game system is one for importing three-dimensional model data for the body of a person such as the user himself or herself or a friend into the virtual three-dimensional space of a computer game and playing that game.

Detail Description Paragraph:
[0388] The game supplier system 1722, in the same manner as the game supplier system 1702 in the game system diagrammed in FIG. 22, provides format information for the three-dimensional models used in various game programs to the modeling server 1721. The store system 1729 and the stereo photographing system 1730, in the same manner as the store system 1005 and stereo photographing system 1006 of the virtual trial fitting system diagrammed in FIG. 8, photograph the body of the user with a plurality of multi-eyes stereo cameras and send those photographed data to the modeling server 1721.

Detail Description Paragraph:

[0389] The user system 1724, which is a personal computer or game computer, for example, has a controller 1727 operated by the user and a display device 1728, and is loaded with a game program for a game wherein human beings appear (such as a fighting game, for example). To the user system 1724, furthermore, if so desired by the user, are connected a plural number (such as 2, for example) of multi-eyes stereo cameras 1725 and 1726, and a stereo photographing program can be loaded for sending the photographed data from those multi-eyes stereo cameras 1725 and 1726 to the modeling server 1721 and receiving three-dimensional model data from the modeling server 1721.

Detail Description Paragraph:
[0392] (2) As indicated in steps S1121 and S1122, the modeling server 1721, upon receiving the game ID and the photographed data for each motion of the user, produces three-dimensional model data for the user's body in the data format for that game ID, for each frame of those photographed data (moving image data) for those motions, by the processing method described with reference to FIG. 17 to 20, and continuously lines up the plurality of three-dimensional model data produced respectively from the series of multiple frames of the images of the motions, according to the frame order. As a result, a series of pluralities of three-dimensional model data configuring the motions is formed. Then, as indicated in step S1123, the modeling server 1721 transmits the series of three-dimensional model data configuring the motions to the user system 1724.

Detail Description Paragraph:
[0393] (3) As indicated in steps S1113 and S1114, the stereo photographing program of the user system 1724, upon receiving the series of three-dimensional model data for the motions, produces a plural number of animation images that look respectively from a number of different viewpoints at the spectacle of the user performing the motions, using the series of three-dimensional models of those motions, and sequentially displays those animated images on the display device 1728. The user thereupon checks whether there are any problems with the series of three-dimensional model data for the motions received. When it has been verified that there are no such problems, the stereo photographing program stores the series of three-dimensional model data for the motions received, and notifies the modeling server 1721 of that receipt.

Detail Description Paragraph:
[0394] (4) As indicated in steps S1124 and S1125, the modeling server 1721, after verifying the receipt by the user of the three-dimensional model data, performs a fee-charging process for collecting a fee from the user, transmits data resulting from that fee-charging process such as an invoice to the user system 1724, and, as indicated in step S1115, the stereo photographing program in the user system 1724 receives and displays those data resulting from that fee-charging process.

Detail Description Paragraph:
[0395] (5) Thereafter, as indicated in step S1116, the user runs the game program on the user system 1724, and the series of three-dimensional model data for the motions stored earlier are used in that game program. For example, in response to control inputs made by the user from the controller 1727, the three-dimensional model 1731 of the user performs the motions of various moves such as straight punches, uppercuts, or tripping, in the virtual three-dimensional space of the fighting game, as indicated in the display device 1728 in FIG. 25.

Detail Description Paragraph:
[0399] The arithmetic logic unit 1744 produces a series of three-dimensional model data, one set after another, at high speed, that move along with and in the same manner, in real time, as the motion of the user 1748, from photographed data (moving image data) from the multi-eyes stereo cameras 1741 to 1743, and send those data to the game apparatus 1745. The game apparatus 1745 imports that series of three-dimensional model data into the virtual three-dimensional space of the game and displays, on the display device 1746, a three-dimensional model 1747 that moves in exactly the same way and with the same form as the actual user 1748. Thus the user 1748 can play the game with the sense of reality that he or she himself or herself has actually entered the world of the game.

CLAIMS:

7. A system for enabling a real physical object to appear in virtual three-dimensional space in a computer application used by a user, comprising: photographed data reception means for

receiving photographed data produced by stereo photographing a real physical object, from a stereo photographing apparatus usable by said user, capable of communicating with said stereo photographing apparatus; modeling means for producing a three-dimensional model of said physical object, based on said received photographed data, in a prescribed data format that can be imported into virtual three-dimensional space by said computer application; and three-dimensional model output means for outputting three-dimensional model data of said physical object, by a method wherewith those data can be presented to said user or to a computer application used by said user.

8. The system according to claim 7, wherein said photographed data received from said stereo photographing apparatus comprise photographed data for a plurality of poses photographed, respectively, when said real physical object assumed different poses; and said modeling means produce said three-dimensional model data for said physical object, based on said photographed data for said plurality of poses, in such a configuration that different poses can be assumed or different motions reproduced.

9. The system according to claim 7, wherein said photographed data received from said stereo photographing apparatus comprise photographed data for moving images photographed when said real physical object performed some motion; and said modeling means produce said three-dimensional model data for said physical object, based on said photographed data for said moving images, in such a configuration that same motion as that performed by said physical object is reproduced.

10. The system according to claim 9, wherein said modeling means produce said three-dimensional model data so that same motion is reproduced as the motion performed by said real physical object, following said latter motion substantially in real time during the photographing by said stereo photographing apparatus.

11. A method for enabling a real physical object to appear in virtual three-dimensional space in a computer application used by a user, comprising the steps of: receiving photographed data produced by stereo photographing a real physical object from a stereo photographing apparatus that can be used by said user; producing a three-dimensional model of said physical object, based on said photographed data received, in a prescribed data format capable of being imported into virtual three-dimensional space by said computer application; and outputting three-dimensional model data for said physical object by a method that enables the date to be provided to said user or to said computer application used by said user.

12. A system for enabling a real physical object to appear in virtual three-dimensional space in a computer application used by a user, comprising: a stereo photographing apparatus that can be used by said user; and a modeling apparatus capable of communicating with said stereo photographing apparatus, and also capable of communicating with a computer apparatus that can be used by said user; wherein said modeling apparatus has: photographed data receiving means for receiving photographed data produced by stereo photographing a real physical object from said stereo photographing apparatus; modeling means for producing a three-dimensional model of said physical object, based on said photographed data received, in a prescribed data format capable of being imported into virtual three-dimensional space by said computer application; and three-dimensional model transmission means for transmitting three-dimensional model data for said physical object to said computer apparatus that can be used by said user.

13. A system for enabling a real physical object to appear in virtual three-dimensional space in a computer application used by a user, comprising: a computer apparatus for execution of said computer application by said user; a stereo photographing apparatus that can be used by said user; and a modeling apparatus capable of communicating with said stereo photographing apparatus and said computer apparatus; wherein said modeling apparatus has: photographed data receiving means for receiving photographed data produced by stereo photographing a real physical object from said stereo photographing apparatus; modeling means for producing a three-dimensional model of said physical object, based on said photographed data received, in a prescribed data format capable of being imported into virtual three-dimensional space by said computer application; and three-dimensional model transmission means for transmitting three-dimensional model data for said physical object to said computer apparatus.

14. A method for enabling a real physical object to appear in virtual three-dimensional space in a computer application used by a user, comprising the steps of: stereo-photographing a real

physical object; producing a three-dimensional model of said physical object, based on photographed data of said physical object obtained by stereo photographing, in a prescribed data format capable of being imported into virtual three-dimensional space by said computer application; and inputting three-dimensional model data for said physical object into a computer apparatus capable of executing said computer application.

Previous Doc Next Doc Go to Doc#

L39: Entry 7 of 12 File: PGPB Mar 21, 2002

DOCUMENT-IDENTIFIER: US 20020035451 A1
TITLE: Spatail data enabled engineering, construction, and operations computer-aided design (CAD) project system, method and computer program product

Abstract Paragraph:
A system and method of geo-spatially viewing project oriented data relating to engineering, construction, and operations (ECO) computer-aided design (CAD) services online using a browser-based viewer can include a three-tiered client/server architecture for accessing a database storing a plurality of files, each of which is stored as a series of components including spatial information. In an exemplary embodiment of the invention, a system, method and computer program product includes a browser-based project-oriented, collaboration system including a viewer operative to enable viewing of at least one of a business, a project, and a person, geospatially according to geospatial attributes associated with the business, project or person.

Summary of Invention Paragraph:
[0005] CAD software is well known, and is used by architects, engineers, artists, and the like to create precision drawings and technical illustrations. CAD software is used to create two-dimensional (2-D) drawings and three-dimensional (3-D) models. Applications such as, e.g., MicroStation.TM. products, which are developed by Bentley Systems, Inc., Exton, Pa. U.S.A., and AutoCAD.RTM. products, which are developed by Autodesk, Inc., San Rafael, Calif., U.S.A., are typical of such CAD software, which may be used in the engineering, construction, and operations (ECO) marketplace. Also illustrative of the systems and methods used in the ECO marketplace are: U.S. Pat. No. 5,815,415 for a "Computer System for Portable Persistent Modeling"; U.S. Pat. No. 5,987,242 for an "Object-Oriented Computerized Modeling System"; and co-pending U.S. Ser. No. 09/296,738, filed Apr. 22, 1999, for a "System for Collaborative Engineering Using Component and File-Oriented Tools", each of which is commonly assigned to the assignee of the present invention, and is incorporated herein by reference.

Summary of Invention Paragraph:
[0014] HTML is perhaps the most portable of today's markup languages. It is supported by over 100 million Web browsers, and has become the de facto standard for transmitting information between people. HTML has many advantages, not the least of which is that HTML browsers are typically free. They are very powerful, have a combination of third-party add-ins and server-side content support, and a vast range of information is being delivered to HTML browsers. HTML document browser interfaces are easy to build into existing products because of the simplicity of HTML. HTML is easy to learn because it is very simple. There are only a couple dozen tags, but less than half of them are used in most situations. In working with HTML for just a few years, it has become very evident to users that the hypertext model really does work across systems that are otherwise unrelated. Any page can link to any other publicly accessible page simply by entering the address. There are some specialized structures in HTML, but the structures are mostly used to effect a certain formatting look. Because of the simplicity and low cost of HTML, a huge information base has been formed that makes HTML even more valuable.

Summary of Invention Paragraph:
[0015] HTML's simplicity, while making it valuable as a basic way of delivering simply structured information, causes it to fall short of being a long-term method of delivering complex information types. HTML is a very weak presentation tool that lacks even the most fundamental page-oriented formatting capabilities, such as, e.g., hanging indents, white-space control, justification, kerning, and hyphenation. HTML does not handle multiple-column snaking very well, either. However, because of the nearly universal compatibility, web site designers are getting around these problems by using tables to fake multiple columns and indents, GIF graphics to create certain designs with type and white space, and other such machinations. In

such cases, HTML itself has simply become a shell that contains the real markup.

Summary of Invention Paragraph:
[0018] This balkanization of HTML has made webmasters and users frustrated and looking for a better solution. One way the HTML keepers are trying to extend HTML is by providing the ability to create more customized styles, while keeping the same markup. This is being done by cascading style sheets (CSS), which is a technical recommendation by the W3C (World Wide Web Consortium). CSS separates structure (i.e., HTML markup) from format (i.e., how it looks). While this is a good step, users are still stuck with the basic HTML tag set.

Summary of Invention Paragraph:
[0019] HTML's predecessor, the Standard Generalized Markup Language (SGML), is an international standard that is more than ten years old. It was originally designed to provide a way of describing text-based information so organizations could interchange information easily. Since then, SGML has become valuable in describing information sets so companies can get beyond the restrictions of paper-only publishing. SGML provides a way of creating markup languages customized for each document type, and separating the content from eventual formatting.

Summary of Invention Paragraph:
[0023] Another document format, the "portable document format" (PDF), was developed by Adobe Systems Inc., in order to provide a system-independent way of delivering page-based information. PDF files are created by printing to a PDF driver or by so-called "distilling" a PostScript file. The resulting PDF file can be read using a tool from Adobe called "Acrobat Reader," which is freely available on most popular systems in use today.

Summary of Invention Paragraph:
[0024] PDF provides electronic pages with impressive page fidelity. Type, graphics, and color are all reproduced as they are on paper. Even hot links and other electronic object types, like movies and sounds, can be added to a PDF file. PDF files are inexpensive to create, and are used by many companies to deliver page-formatted information without the high cost of postage. Since the end user gets something that looks very much like paper, training costs are low.

Summary of Invention Paragraph:
[0025] However, PDF creates large files with little structural information. PDF files are not nearly as flexible as other electronic formats because the main goal is to recreate a paper page, and not to provide a way of delivering intelligent document structure to a user. There is also little support for searching, although Adobe has other products that can index many different PDF files for cross-document searching and navigating (i.e., "turning" pages, "flipping" from section to section, and "scanning" the page for text of interest). Other than this limited support, navigation is limited. Another problem is page fidelity. PDF pages are not necessarily pixel-by-pixel replicas of a page that might be printed by the owner of the document. This is partly due to the fact that the fonts which are used to create the document originally might not be on the machine that eventually views the document. It was found desirable, therefore, to come up with a solution having the low cost and simplicity of HTML, the power and flexibility of SGML, and the pleasing formatting capabilities of PDF. Proponents of the "Extensible Markup Language" or XML, believe that standard fits the bill.

Summary of Invention Paragraph:
[0027] aecXML is a framework for using the XML standard for electronic communications in the architectural, engineering and construction (aec) industries. It includes an XML schema to describe information specific to the information exchanges between participants involved in designing, constructing and operating buildings, plants, infrastructure and facilities. Various software applications used by aecXML participants can transfer messages formatted according to the aecXML schema to coordinate and synchronize related project information. Additionally, its specification facilitates e-commerce between suppliers and purchasers of equipment, materials, supplies, parts and services based on that same technical information. It would also be desirable, therefore, to provide CAD software usable over the Internet by way of dynamically rendered web pages using an extensible markup language, such as XML.

Summary of Invention Paragraph:
[0028] Another significant drawback to project hosting websites such as Project Point is that they are limited to proprietary drawing file formats created by AutoCAD (e.g., the DWG and DWF file formats). As a result, all of the users accessing such websites must run AutoCAD, or they

will not be able to participate in the project. It would, therefore, be desirable to provide CAD software for an Internet environment, which is flexible and adapted to multiple drawing file formats. DWF files, for example, suffer from the disadvantage that they remove the nonvisual components of a drawing, such as attributes, properties, and complex object behavior. Moreover, the DWG format is an electronic plot that is view only, so that no matter how many people access the document, the integrity of the design is preserved. DWF files support hyperlinks, and such hyperlinks are usually not limited to data files. They can also activate Java applications, scripts, or other DWF files.

Summary of Invention Paragraph:
[0032] Component-level information management leads to better coordination of hybrid projects. That is, the concepts of component-level information management apply not only to MicroStation, but also to other types of engineering files (e.g., those created by AutoCAD. AutoCAD-specific workflows will, thus, gain the same benefits of simultaneous write access and project history as do MicroStation workflows. In addition, since many engineering projects involve hybrid data sources, and an amalgamation of both AutoCAD DWG and MicroStation DGN files in the same project, the concepts of component-level information management will allow these hybrid file formats (and, in turn, hybrid workflows) to come together in one efficient and coordinated environment.

Summary of Invention Paragraph:
[0033] Conventional project oriented CAD systems and methods only provide textual based information regarding projects, people and businesses. Such textual information is difficult to analyze. It is desirable that a more user-friendly, graphical display means be used to convey the project-oriented data information. For example, conventional CAD systems lack geospatial visualization tools or geographic information system (GIS) viewing functionality.

Summary of Invention Paragraph:
[0043] An exemplary embodiment of the present invention includes a browser-based, geospatial viewer of information relating to a project-oriented ECO CAD collaboration system.

Summary of Invention Paragraph:
[0044] Advantageously, the viewer of the present invention can provide a geospatial visualization of project collaboration datasets by spatially enabling the project collaboration datasets, by adding a spatial reference to the project collaboration datasets, and by enabling browser-based viewing of the project-oriented collaboration system data via a map.

Summary of Invention Paragraph:
[0052] In an exemplary embodiment of the present invention, a Java viewer including a pure Java viewing application, is tied to the project-oriented collaboration system such as, e.g., Bentley's Viecon, to access live geospatial data. The Java viewer allows users to display and manipulate data quickly and easily.

Detail Description Paragraph:
[0070] Block diagram 100 can further include a plurality of application servers 112a, 112b, coupled to the web servers 110a, 110b. In an exemplary embodiment, a load balancer 108b is shown balancing workload across application servers 112a, 112b. In an exemplary embodiment of the present invention, there can be six such application servers. Each of the application servers 112a, 112b, like the web servers 110a, 110b, in an exemplary embodiment, can include a Dell PowerEdge 2450 server, configured with a 733 MHz Pentium III processor, 256 MB RAM, and dual, mirrored 9.1 GB fixed disk drives. In an exemplary embodiment, each of the application servers 112a, 112b can further include a Microsoft Windows NT operating system, and a Total-e-Business.TM. platform, developed by Bluestone Software, Inc., Philadelphia, Pa. U.S.A., and including the Total-e-Business Server (formerly known as "Sapphire/Web"). Bluestone's Universal BusinessTM Server, Release 7.0, for example, can, in an exemplary implementation embodiment, be used to manage the ECO CAD system and methods of the present invention, while running on each of the application servers 112a, 112b. In an exemplary embodiment, load balancers 108a, 108b can include Bluestone's Load Balance Broker (LBB) loaded on each of the web servers 110a, 110b, to facilitate balancing of the load of communications between each of the web servers 110a, 110b and each of the application servers 112a, 112b.

Detail Description Paragraph:
[0083] FIG. 2 depicts an exemplary embodiment of a GUI 200 creating a new business according to

the present invention. The geospatial viewer of GUI 200 includes various exemplary buttons 204, 206, 208, 210, 212, 214, 216 and 218, and geospatial location coordinates 224, 226. It will be apparent to those skilled in the art that additional buttons and/or coordinates can be included within the spirit and scope of the invention. The map includes business 122.

Detail Description Paragraph:
[0094] In an exemplary embodiment of the present invention, the geospatial enabled ECO CAD application according to an exemplary embodiment of the present invention can include, e.g., two separate computer programs: (1) a server program on a server side 112a of the application; and (2) a client program on a client 102 side of the application. The two programs can be loaded into memory and executed on a single computer; or they may be loaded on two different computers connected together by a computer network; or they may be combined into a single program. In an exemplary embodiment of the present invention, the geospatial project-oriented viewer is Internet enabled, or web-enabled, including a JAVA applet or pure JAVA viewer which can in one embodiment be browser-based. The spatially enabled project collaboration system of the present invention, can provide a backoffice tool set which can be used by a user to populate the project database with geospatial location attribute 118 information.

Detail Description Paragraph:
[0101] The Plot module can provide a means of viewing drawings over the Internet. Integrated with any MAPI compliant E-mail system, such as Microsoft Exchange, the plot module can allow users to E-mail electronic plots to members of the user's extended work team, such as customers and suppliers who do not have access to MicroStation or AutoCAD. The interchange module can enable CAD administrators in charge of a workgroup's project data to manage various file formats. Interchange can simplify the process of maintaining synchronized design data by automating the conversion of between files of different formats. The review module can enable simple, accurate, standalone viewing by project reviewers. Team members who require design review capabilities can do so without altering design data and without purchasing additional software licenses. A user can extend the use of live engineering project drawings and models throughout the enterprise without affecting the user's bottom line. Moreover, team members can use this viewing tool for the entire life of the project.

Detail Description Paragraph:
[0102] A user of the plot module merely defines the plot area using a view or fence. Then, the user can compose a recipient list using Internet E-mail addresses, local mailing lists, or a combination of both. A short note may then be added to the E-mail. Finally, the user can send the eplot to a recipient as illustrated in FIG. 7. A plot publisher contained within the plot module can automatically create a compact viewable electronic plot file, which can then be sent to one of the application servers 112a, 112b. At the same time, the plot module can also send out an E-mail message to everyone on the user's recipient list. This E-mail can inform the recipients that they have received a Eplot, and can provide the recipients with the Internet location (URL) where the eplot can be viewed. There are no complex plug-ins or viewers to install. The recipient simply clicks on the link in his/her E-mail, automatically connecting to the plot module in the browser using a compact yet powerful Java viewer.

Detail Description Paragraph:
[0103] Using a viewer portion of the plot module, users can accurately measure distances, zoom in and out on various areas of the design without degradation of detail and change level or layer settings. A publishing client of the plot module, in an exemplary embodiment of the present invention, can include: (1) MicroStation/J V07.00.01.11 or later; (2) Internet Explorer V4 or later; and (3) for address book support, a MAPI complaint E-mail system is required. Otherwise, if no MAPI compliant E-mail system is present, the publisher can be limited to using Internet E-mail addresses. A viewing client of plot module can include: (1) Internet Explorer V4 or later; or (2) Netscape Navigator V4.08 or later. Further details regarding the plot module 205 can be found in U.S. Ser. No. 09/586,786, filed concurrently herewith, for a "System, Method and Computer Program Product for Generating a Viewable Computer Aided Design (CAD) Eplot" (Attorney Docket No. 36488-160032).

Detail Description Paragraph:
[0104] CAD administrators in charge of a workgroup's project data often deal with various file formats. The interchange module can simplify the process of maintaining synchronized design data by automating the conversion of between files of different formats. Interchange can run in the background and can search a specified source folder/directory and its

subfolders/subdirectories for input in one format (e.g., MicroStation design files) and can convert any designs updated since the last time the interchange module checked to a different format file (e.g., AutoCAD) in a target folder/directory. The interchange module can compare the date and time of all of the source files against their corresponding destination counterparts. A database can be continually maintained in the disk arrays of SAN 114 with information on source files and their dependencies, such as MicroStation BASIC settings files used for data conversion, attached reference files, etc. It can automatically create a new destination file when either the source file or one of its dependencies is changed. Source files that are already synchronized with their respective destination file are left alone.

Detail Description Paragraph:
[0105] In an exemplary embodiment of the invention, the interchange module can support the following combinations of design data: (1) DGN to DWG; (2) DGN to DXF; (3) DWG to DGN; (4) and DXF to DGN. Interchange is a standalone utility. That is, downloading and installation on a host computer is required. Further details regarding the interchange module can be found in U.S. Ser. No. 09/586,786, filed concurrently herewith, for a "System and Method to Maintain Real-Time Synchronization of Data In Different Formats" (Attorney Docket No. 36488-160120).

Detail Description Paragraph:
[0110] By entering basic contact information, project team members obtain first-level access to the site. This information can include geospatial location information such as the address of the team member. This gateway is also the first level of security, so that access to a user's proprietary design data is limited to only those users who are authorized to see the data. Following the registration process, the user is given an assigned passwords and other permissions that can provide unique project-level identification and grant the user access to the suite of tools on the site.

Detail Description Paragraph:
[0114] All of the Document Tools can be designed to provide a continuous flow of information. The user can post projects by adding information, attaching drawings, starting discussions, or adding milestones. The project posts can be available at both the Project and User level. The user can also view project data. Users without CAD system access and experience can view and review the CAD data. Redlining tools can focus discussion on the precise areas of concern; for example, does this duct pass through a wall? Will the electrical conduits fit in the given space? The user can also use text-based tools, because threaded discussions can provide a continuous stream of information in a purely interactive model.

CLAIMS:

1. An online system for providing engineering, construction, and operations (ECO) computer-aided design application services, comprising: a computer network, including server means and a plurality of clients each of which is adapted to be connected to said server means over said network; means for operating said server means and said plurality of clients, said operating means supporting a runtime environment for the ECO application on said network; graphical user interface means adapted to be displayed on said plurality of clients, said graphical user interface means including a plurality of personalized spaces; a database storing a plurality of files, including engineering documents and drawings, with a plurality of different file formats; a plurality of collaborative modules, each of which is adapted to be run over said network, said collaborative modules including: means for accessing said engineering documents and drawings; and means for managing an engineering document; and means for viewing at least one of a business, a project, a task, a document, and a person, geospatially according to geospatial attributes.

2. The system according to claim 1, further comprising: means for translating drawings of one of said plurality of file formats to another of said plurality of file formats; means for mirroring said database drawings of said one of said plurality of file formats with said other of said plurality of file formats; and means for sharing translation settings used by said translating means.

10. A method of providing application services for an engineering, construction, and operations (ECO) application, comprising the steps of: providing a computer network, including one or more servers, each of which is adapted to be accessed by a plurality of clients over said network; providing means for operating said one or more servers, which supports a runtime environment

for the ECO application on said network; providing a graphical user interface which is adapted to be displayed on said plurality of clients, said graphical user interface including a plurality of personalized spaces; providing a database for storing a plurality of files, including engineering documents and drawings, with a plurality of different file formats, each of said plurality of files being stored in said database as a series of components; providing a plurality of collaborative modules, each of which is adapted to be run over said network, said collaborative modules including: means for accessing said engineering documents and drawings; and means for managing an engineering document; controlling access to the ECO application such that only those clients authorized to use the ECO application for a given project are permitted access to said project; and means for viewing at least one of a business, a project, a task, a document, and a person, geospatially according to geospatial attributes.

14. A computer-readable medium, comprising: a first code segment for providing a graphical user interface to a database storing a plurality of files, including engineering documents and drawings, with a plurality of different file formats, each of said plurality of files being stored as a series of components; a second code segment for accessing said engineering documents and drawings; a third code segment for managing an engineering document; a fourth code segment for managing a discussion; a fifth code segment for managing a form; a sixth code segment for managing time; and a seventh code segment means for viewing at least one of a business, a project, a task, a document, and a person, geospatially according to geospatial attributes.

16. The computer-readable medium according to claim 14, wherein said second code segment further comprises: means for translating drawings of one of said plurality of file formats to another of said plurality of file formats; means for mirroring said database drawings of said one of said plurality of file formats with said other of said plurality of file formats; and means for sharing translation settings used by said translating means.

19. A system for viewing project-oriented data relating to engineering, construction, and operations (ECO) computer-aided design (CAD) services online comprising: a browser-based geospatial viewer operative to enable geospatial viewing of data according to geospatial attributes associated with said data; and a browser-based project-oriented, ECO CAD collaboration system.

Previous Doc Next Doc Go to Doc#

WEST Search History

Hide Items Restore Clear Cancel

DATE: Tuesday, January 04, 2005

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR	
☐	L39	L38 and (design near data)	12
☐	L38	L36 and (format$)	120
☐	L37	L36 and (format$ near viewer$)	1
☐	L36	L34 and viewer	173
☐	L35	L34 and (libray near format)	0
☐	L34	(CAD near system) and (three near dimensional)	2249
☐	L33	L32 and verifier	1
☐	L32	L31 and (library near format)	4
☐	L31	(application near software) and (CAD near system)	251
☐	L30	view$ near design near software	6
☐	L29	view$ near intelligent near design	1
☐	L28	L27 and (program$ near interface)	0
☐	L27	L26 and properties	2
☐	L26	L25 and format	2
☐	L25	L24 and (data near model)	2
☐	L24	L22 and (verifier)	11
☐	L23	L22 and (libray near format)	0
☐	L22	L21 and (intelligent near design)	14
☐	L21	716/$.ccls.	8207
☐	L20	L19 and (library near format)	0
☐	L19	(software near design) and (import$ near application)	234
☐	L18	L1 and l15	0
☐	L17	L15 and (verifier)	3
☐	L16	L15 and (format near verifier)	0
☐	L15	(library near format) and (software near design)	26
☐	L14	L12 and (library near format)	0
☐	L13	L12 and verifier	0
☐	L12	l8 and (software near design)	4
☐	L11	L10 and format	8
☐	L10	L9 and (data near model)	10
☐	L9	L8 and design	87
☐	L8	(import near application)	172
☐	L7	(intelligent near design) and (import near application)	1

☐	L6	L4 and (import near application)	1
☐	L5	L4 and (import$ near application)	5
☐	L4	L3 and match$	125
☐	L3	L1 and (format$)	188
☐	L2	L1 and (plurality near format$)	0
☐	L1	intelligent near design	335

END OF SEARCH HISTORY

L11: Entry 7 of 8 File: USPT Jun 22, 2004

DOCUMENT-IDENTIFIER: US 6754885 B1
TITLE: Methods and apparatus for controlling object appearance in a process control configuration system

Parent Case Text (2):
This application claims the benefit of priority of U.S. Ser. No. 60/134,597, filed May 17, 1999, entitled INTEGRATED DESIGN AUTOMATION CONTROL ALGORITHM CONFIGURATOR ARCHITECTURE, the teachings of which are incorporated herein by reference.

Brief Summary Text (9):
The principal function of controllers is executing control algorithms for the real-time monitoring and control of devices, processes or systems. They typically have neither the computing power nor user interfaces required to facilitate the design of a control algorithm. Instead, the art has developed configurators. These are typically general purpose computers (e.g., workstations) running software that permit an engineer or operator to graphically model a device, process or system and the desired strategy for controlling it. This includes enumerating field devices, control devices, controllers and other apparatus that will be used for control, specifying their interrelationships and the information that will be transferred among them, as well as detailing the calculations and methodology they will apply for purposes of control. Once modeling is complete and tested, the control algorithm is downloaded to the controllers.

Detailed Description Text (47):
In the illustrated embodiment, the minimum attribute set for a Parameter Definition is as follows: Name The unique identifier for accessing the parameter within a Parameterized Object. There cannot be more than one parameter in a Parameterized Object with the same name. This is the name used when downloading the parameter to a target machine. Group A list of Parameter Groups which this parameter belongs to. Label An internationalizable string used to label the Parameter in the user interface. Data Specifies the data type of the Parameter. Integer, float, boolean, and Type string are examples of a data type. Depending on implementation, the length of the data can be either an attribute of the data itself or of the Parameter Definition. Can also be implemented via sub-classes of Parameter Definition. Behavior Specifies a set of behaviors the Parameter exhibits. Examples include whether the parameter could be edited or associated with another Parameter. This can be implemented as a bitmask. Help Specifies internationalizable help associated with the particular Parameter Definition. The help consists of both a verbose and terse version. The verbose version is used by the help system and the terse version is used for such things as short messages and tool tips. Edit Specifies a specific control type used to edit the value attribute Control associated with the Parameter Definition. This edit control type is Type used by any application editing this parameter, whether it is displayed in a property sheet, or in a spreadsheet format. Range Specifies a range of valid values for the Value attribute. Value Specifies the value of the Parameter. This value is type specific which is specified by the type attribute. Formula Provides a placeholder to contain the user-provided formula for Parameter Definitions which have their Value attribute determined by a formula. Format Specifies a C-printf type specification for displaying the value attribute.

Detailed Description Text (56):
The Parameter Override object is used by a Parameterized Object to override attributes of inherited parameters other than value, high range, and low range. Attributes which are typically overridden using this object include which parameter groups a parameter belongs to, format, and help strings.

Detailed Description Text (72):

FIG. 11 provides a depiction of the Parameter Definition Editor. The Framework automatically provides the menu and toolbar services which the editor may need, a tabbed tree pane (shown on the left side of the figure), and a generic view pane which the application programmer can use for just about anything--e.g., a graphical "canvas", or a grid control able to display data in a spreadsheet-like format.

Detailed Description Text (81):
Consequently, it is convenient to think of object types as being contained within a type hierarchy. Each branch in the hierarchy would correspond to an object type category, whereas the leaves, or endpoints, of each branch would correspond to specific object types. The remainder of this section will present the data model, with examples, of this type hierarchy for IDA.

Detailed Description Text (89):
All instances in the Object Type hierarchy are able to act as collections of Typed Objects. That is, each Object Type is able to maintain a list of all Typed Objects which are directly associated with the type itself. For example, all instances of an AIN block will contain a pointer back to the AIN instance of Object Type. In turn, the AIN instance of Object Type will maintain a list of all instances of AIN blocks in the configuration. This containment is meant to be only one level deep--in other words, there is no need for I/A Block, the containing instance of AIN, to also maintain a list of all AIN blocks (although nothing in the design would prevent it, if desired).

Detailed Description Text (133):
All other subclasses within the System Hierarchy simply represent another view of existing configuration components. For example, a Network Hierarchy could display a view of the configuration from a System Definition point of view, showing a hierarchy of networks, nodes, stations, FBMs and other hardware. Since the only grouping of configuration objects in the current design is by object type, these subclasses have to use the relationships specified in the Connections discussion in order to know what to display (i.e., by network, by location, etc.).

Detailed Description Text (281):
This design allows object differentiation between the slots. For example, it is easy to model (in the parameter definitions) the fact that the first two slots of a cell (represented by the first two parameter overrides) can only accommodate a power supply, and the remaining slots available for modules.

Detailed Description Text (434):
Graphical Objects require the management of graphical characteristics such as line weight, line style, line color, fill color, etc. These properties may vary depending upon the object type being modified, and are displayed on that object's property pages appropriately when edited. Annotators, or text strings, support the use of macro substitution, allowing the user to enter specific, predefined macros embedded within text strings. When the text string is displayed on the Sheet Template, the macro is replaced with the appropriate value. Examples of things for which macros are defined include: report name; customer name; user name; and date and time in various formats.

Detailed Description Text (484):
A number of Standard Report Templates are included. These include implementation-standard Graphical and Tabular Report Templates that are provided as read-only templates. With these templates, users are able to report data for any objects in a number of practical formats. Users also use these Standard Report Templates to build custom Composite Report Templates.

Detailed Description Text (495):
The Crystal Reports report expects to find data in a fixed "n x n" array. It uses this data to populate itself in its persisted format.

Detailed Description Text (751):
Objects may be indirectly associated with a Process Area. For example, a Block may be associated with a specific Process Area as the result of being part of a Loop or Compound associated with that area. The ramifications of this on the design needs to be explored further.

Detailed Description Text (795):
This section describes major functions of the configurator architecture, including sample user interface representations. Note that these user interface sample are intended to illustrate functionality described in the text and are not intended to show all design details, such as menus and toolbars. In the object model diagrams included in this section, shading indicates classes supplied by the IDA Framework and a dashed line border indicates classes described in another subsection of this document.

Detailed Description Text (988):
Database functions Create and edit Composite Block Definitions, Composite Blocks, Loop Templates, Template-Derived Loops, Simple Loops Create instances of Blocks and Composite Blocks on a control algorithm diagram Connect parameters between blocks on diagram Move connections Generate Display file for loop or composite block Define parameters to display in block display, source and sink regions Enter diagram title, description, info, local block names Define general formulas used for all blocks in diagram Attach Modifier blocks to blocks in diagram or entire object Add, delete blocks to sheet Edit block parameters via property sheets Copy blocks from one instance of editor to another or within single editor via drag-drop. Create new loops, composite block definitions, loop templates from groups of selected blocks Maintain version history and audit trail of templates, definitions, instances Security based on default and administrator-configurable read, write, download, etc. permissions, as provided by Framework security functions Allow only authorized developers to modify implementation standard Block Definitions and Loop Templates Assign Blocks in Loops to different Compounds or all Blocks in Loops to a single Compounds Define block processing order for blocks in a loop. This value is a suggested order which can be overridden by actions in the Compound Editor. Determine Block names at Loop instantiation time based on name macros, loop macros, and Modifier Blocks applied Download Blocks/Loops to CP Provide "Save As" functionality--Composite Block can be saved as Loop, vice-versa Ensure valid connections between blocks Assign Blocks to Compounds either individually or by Loop Provide back documentation capability in support of Import/Export Manager. This includes the ability to generate a default drawing layout for Loops and Loop Typical (templates) imported from FoxCAE. Provide bulk Loop generation capability in support of Tag List Editor capability to generate Loops from the Tag List. Import and export Blocks, Block Definitions, Composite Block Definitions, Loops, Loop Templates, Template-Derived Loops to/from diskette using IDA format

Detailed Description Text (1115):
The object model, as described above, provides Template-derived Loops and Composite Blocks with their own Control Algorithm Diagram Document objects. This allows more flexibility for adding Modifier Blocks and for repositioning Blocks defined in the definition objects. An alternative approach to consider during detailed design is to have Template-derived Loops and Composite Blocks use the document objects associated with the Loop Templates and Composite Block Definitions, instead of having their own documents. Class Relationships:

Detailed Description Text (1149):
The Ladder class is the top level container for a PLB ladder. It contains an ordered set of connections to Ladder Rungs. The description, last modify date, compiled binary version of the ladder, and last compile date are all maintained as parameters of a Ladder. Only successfully compiled Ladders are stored to the database for later download. The last modify date and last compile dates can be compared before downloading as a validity check. TechIDs and their user-defined descriptions are maintained by this object. TechID descriptions can be modified in the context of any symbol on a ladder diagram. When the description is changed for one symbol, every symbol referencing the same TechID in the same ladder reflects the change. This class provides an interface for basic syntax checking for the ladder. It verifies that the ladder has a valid number of rungs (>0) and queries the rungs for validity. This class has the ability to generate a ladder source in an appropriate format for the existing ladder compiler.Class Relationships:

Detailed Description Text (1155):
A Ladder Rung is the only component which can be connected to a Ladder. The Ladder Rung maintains connections to an ordered set of Ladder Lines. It also contains a string parameter which acts as a rung descriptor. A Ladder Rung consists of a primary line followed by zero or more secondary lines. The primary line consists of Ladder Elements connected from the left power rail to the right power rail in the diagram. Secondary lines can supply additional logic

via "or" connections to the primary line. The top most line is preferably the primary line. This class provides basic syntax checking for the ladder rung. It verifies that the rung has a valid number of lines (>0) and queries the lines for validity, ensuring that the first line meets the criteria for a primary line. This class can provide data in the file format necessary for the existing ladder compiler. It supplies data specific to the rung and invokes similar methods on the contained lines. Class Relationships:

Detailed Description Text (1161):
The Ladder Line object represents one line of a rung in a ladder logic diagram. One or more Ladder Lines comprise a Ladder Rung and one or more Ladder Rungs comprise a Ladder. A Ladder Line can either represent a primary ladder line or a secondary ladder line. A primary ladder line is the logical first line of a rung and is indicated by a power connection from left power rail to right power rail. All connections to elements on secondary lines preferably branch from or join the primary line. Using the Ladder Line Parameter Connection Type Specifier, this class determines what types of elements can be dropped into a given slot on the line. Only Output Ladder Elements are allowed in slot 8 and only Input Ladder Elements are allowed in slots 1-7. This class provides syntax checking, based on whether it is a primary or secondary line. If this line is the first in a rung, Ladder Rung can query this class to validate that it is a primary line. For all other lines in a rung, this class would be queried by Ladder Rung to determine if it were a valid secondary line. This class can provide data in the file format necessary for the existing ladder compiler. It supplies data specific to the line and invokes similar methods on the contained elements. Class Relationships:

Detailed Description Text (1265):
For the purposes of the following discussion, "tag list" refers either to the generic concept of tag lists or to the externally generated tag list being imported and exported from IDA while "Tag List" refers only to the IDA database object of that name. A tag list is a collection of data describing the physical I/O, its labeling, its connection with the DCS (Distributed Control System) and the loops involved in controlling the process using the i/o. It is sometimes the output from (or the input to) a Plant Design System, such as the InToolS.TM. product of Intergraph Corporation. Since changes to this data can occur either within IDA or within the Plant Design System, it is necessary to be able to exchange this data in a suitable form. The most common format for data exchange is delimited ASCII files and this is one medium used by IDA. In addition, IDA can import and export tag list data in DBF format for compatibility with FoxCAE.

Detailed Description Text (1271):
Like Blocks, Tag List Rows are homogeneous collection of Parameterized Objects where each Tag List Item within a row is a Parameter and the Parameter name defines the "column" name. For purposes of importing, editing and exporting tag lists, the Tag List is presented to the user in a data grid format where each Tag List Row is presented as a row in a data grid and each column represents a Tag List Item. This representation is used within the Tag List Editor.

Detailed Description Text (1276):
For use in importing and exporting external databases, a dialog box based interface is used. Shown in the figures are examples of dialogs for mapping fields between ASCII input files or database tables and the tag list "fields". The displays shown FIG. 103 et seq., include those typically used with Plant Design Systems in general and InTools in particular (to preserve the existing FoxCAE functionality, InTools compatibility is a requirement). The dialog box shown in FIG. 103 is used to map an ASCII, delimited list into the appropriate Tag List Items. Once a text (ASCII) file is chosen, a count of the number of delimited fields per line is made and the input file is verified to contain exactly that many fields in every line. Next the upper left list in the dialog is generated, showing one line per field, numbering each field and showing the content of each field in the first line of the file. The upper right list shows the attributes of a each Tag List Item in a Tag List Row.

Detailed Description Text (1278):
FIG. 104 shows the selections used in the illustrated embodiment to export a Tag List in delimited ASCII format. In this example all of the Tag List Items have been assigned (mapped) to fields in the text output file. The human interface components have the same actions as described in the previous figure with the exception that the "Export" button causes the creation of the text output file.

Detailed Description Text (1280):
An interface for defining the Tag List Row structure by specifying the Tag List Items and their positions within the row is used. This interface is identical to that used by the Block Definition Editor since the Tag List Items are Parameter Definitions and the Tag List Row specification is actually a defining Block. The Tag List Editor will share the same dialog classes used to implement this functionality in the Block Definition Editor, with methods overridden, where necessary, to enforce the restrictions required in Tag List construction. As a result, the detailed interface description is the same as in the Block Definition Editor. It is not necessary that every Tag List Item be used within IDA. Some fields in external tag lists are used for documentary purposes by the Plant Design System using the list and may not be applicable to IDA. However, to preserve an interchangeable format with a Plant Design System, fields which are not accessed by IDA should still be defined as Tag List Items so that the export functionality preserves the Plant Design System fields intact. Users can add Tag List Items to the Tag List Definition, but only authorized developers are allowed to deleted or modify the implementation standard items.

Detailed Description Text (1352):
The Download Agent contacts the Download Service to obtain a reference for an ICCAPI object to use for each Control Station. The Agent formats ICCAPI instructions, based on the action records in its optimized list, and invokes the appropriate method of the ICCAPI object to process the actual download to the physical Control Station.

Detailed Description Text (1411):
The Download Service for OI downloads is comprised of two applications. The first application is a generic application which replicates data files, created by the Download Agents, to a specified list of targets. The second application is specific to OI applications and performs the needed functionality to import the replicated data into the OI application. The OI import application is specific to each OI application. Responsibility for the implementation of the import services resides with the OI application team. The Download Agent may interact directly with an OI Application Server to import the data, without invoking replication services, if appropriate.

Detailed Description Text (1594):
As depicted in FIG. 119, as each editor is activated via an object selection (or some other means), it becomes the "active" Document Template for the particular editor frame. Only one Document Template can be "active" at a time. Each Document Template knows how to deal with the objects it is able to edit. The editors may present an edit capability to the user in a graphical format, in a grid (spreadsheet-like) format, or other appropriate format depending upon the object being edited, and the context it is being edited in.

Other Reference Publication (21):
Scharf, Ronald, et al, "Using Mosaic for Remote Test System Control Supports Distributed Engineering," Institute of Computer-Aided Circuit Design--Test and Testsystems Division, University of Erlangen-Nurnberg, Germany, web page print-out from http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/CSCW/scharf/scharf.html (8 pages), 1994.

Other Reference Publication (67):
Gyorki, John R. "PLCs drive standard buses," Machine Design (May 11, 1995), pp. 83-90.

Previous Doc Next Doc Go to Doc#